

PEC 01 · Las ómicas

Nuria Garcia Martinez-Illescas

2024-11-03

Contents

Abstract	1
Objetivos	1
Materiales y métodos	1
Resultados	1
Discusión y conclusiones	12
Repositorio de Github	12

Abstract

Este informe analiza la relación entre la pérdida de masa muscular (cachexia) y el perfil metabolómico de muestras de orina en pacientes. A partir de los datos obtenidos, se estructuraron las muestras usando la clase SummarizedExperiment, y se aplicó un análisis de componentes principales (PCA) para identificar diferencias entre los grupos y metabolitos relevantes. Aunque el PCA no mostró una clara separación entre pacientes con cachexia y controles, observamos que metabolitos como la creatina, el citrato y el hipopurato tienen una influencia significativa en la variabilidad. Estos hallazgos sugieren que dichos metabolitos podrían ser potenciales biomarcadores de cachexia, aunque sería conveniente realizar un análisis adicional para confirmar los resultados: incluyendo la revisión de posibles valores atípicos que podrían estar afectando la interpretación, así como descartar aquellos valores cuya variación es mínima entre grupos favoreciendo así la identificación de los metabolitos decisivos.

Objetivos

El objetivo del presente informe es investigar la relación entre la pérdida de masa muscular (denominada cachexia) y los datos metabolómicos de un conjunto de muestras biológicas.

Materiales y métodos

Los datos analizados en este estudio fueron obtenidos de esta URL sobre un conjunto de muestras de orina de pacientes diagnosticados con cachexia y controles sanos.

Los datos fueron almacenados para su organización en la clase SummarizedExperiment con la finalidad de integrar los datos y metadatos de las muestras. Adicionalmente se realizó un análisis de componentes principales (PCA) para analizar las variaciones entre grupos, y determinar los metabolitos específicos que están asociados a la cachexia.

Resultados

Generación del contendor

Se realiza un análisis del conjunto de datos metabolómicos obtenidos del conjunto de muestras de orina. Comenzamos cargando los datos a partir del archivo CSV, usando la primera fila de nuestros datos como

el título de nuestras filas, lo que corresponde con el ID del paciente. Además reestructuramos la variable Muscle.loss en factores, para facilitar el análisis.

```
human_cachexia <- read.csv("human_cachexia.csv", row.names = 1)
```

```
human_cachexia$Muscle.loss <- as.factor(human_cachexia$Muscle.loss)
str(human_cachexia)
```

```
## 'data.frame': 77 obs. of 64 variables:
## $ Muscle.loss : Factor w/ 2 levels "cachexic","control": 1 1 1 1 1 1 1 1 1 1 ...
## $ X1.6.Anhydro.beta.D.glucose: num 40.9 62.2 270.4 154.5 22.2 ...
## $ X1.Methylnicotinamide : num 65.4 340.4 64.7 53 73.7 ...
## $ X2.Aminobutyrate : num 18.7 24.3 12.2 172.4 15.6 ...
## $ X2.Hydroxyisobutyrate : num 26.1 41.7 65.4 74.4 83.9 ...
## $ X2.Oxoglutarate : num 71.5 67.4 23.8 1199.9 33.1 ...
## $ X3.Aminoisobutyrate : num 1480.3 116.8 14.3 555.6 29.7 ...
## $ X3.Hydroxybutyrate : num 56.83 43.82 5.64 175.91 76.71 ...
## $ X3.Hydroxyisovalerate : num 10.1 79.8 23.3 25 69.4 ...
## $ X3.Indoxylsulfate : num 567 369 665 412 166 ...
## $ X4.Hydroxyphenylacetate : num 120.3 432.7 292.9 214.9 97.5 ...
## $ Acetate : num 126.5 212.7 314.2 37.3 407.5 ...
## $ Acetone : num 9.49 11.82 4.44 206.44 44.26 ...
## $ Adipate : num 38.1 327 131.6 144 15 ...
## $ Alanine : num 314 871 464 590 1119 ...
## $ Asparagine : num 159.2 157.6 89.1 273.1 42.5 ...
## $ Betaine : num 110 245 117 279 392 ...
## $ Carnitine : num 265.1 120.3 25 200.3 84.8 ...
## $ Citrate : num 3714 2618 863 13630 854 ...
## $ Creatine : num 196.4 212.7 221.4 85.6 105.6 ...
## $ Creatinine : num 16482 15835 24588 20952 6768 ...
## $ Dimethylamine : num 633 608 735 1064 242 ...
## $ Ethanolamine : num 645 488 407 821 365 ...
## $ Formate : num 441 252 250 469 114 ...
## $ Fucose : num 337 198 187 407 26 ...
## $ Fumarate : num 7.69 18.92 7.1 96.54 19.69 ...
## $ Glucose : num 395 8691 1353 863 6836 ...
## $ Glutamine : num 871 602 302 1686 433 ...
## $ Glycine : num 2039 1108 620 5064 395 ...
## $ Glycolate : num 685.4 652 141.2 70.8 26.6 ...
## $ Guanidoacetate : num 154 110 183 103 53 ...
## $ Hippurate : num 4582 1737 4316 757 1153 ...
## $ Histidine : num 925 846 284 1043 327 ...
## $ Hypoxanthine : num 97.5 82.3 114.4 223.6 66.7 ...
## $ Isoleucine : num 5.58 8.17 9.3 37.71 40.04 ...
## $ Lactate : num 107 369 750 369 3641 ...
## $ Leucine : num 42.1 77.5 31.5 103.5 101.5 ...
## $ Lysine : num 146.9 284.3 97.5 290 122.7 ...
## $ Methylamine : num 52.5 23.6 18.7 48.9 27.9 ...
## $ Methylguanidine : num 9.97 7.69 4.66 141.17 5.31 ...
## $ N.N.Dimethylglycine : num 23.3 87.4 24.5 40 46.1 ...
## $ O.Acetylcarnitine : num 52.98 50.4 5.58 254.68 45.6 ...
## $ Pantothenate : num 25.8 186.8 145.5 42.5 74.4 ...
## $ Pyroglutamate : num 437 437 713 567 185 ...
## $ Pyruvate : num 21.1 37 29.4 64.1 12.3 ...
## $ Quinolate : num 165.7 73 192.5 86.5 38.1 ...
```

```
## $ Serine : num 284 392 296 1249 206 ...
## $ Succinate : num 154.5 244.7 142.6 144 68.7 ...
## $ Sucrose : num 45.1 459.4 160.8 111 75.2 ...
## $ Tartrate : num 97.51 32.79 16.28 837.15 4.53 ...
## $ Taurine : num 1920 1261 4273 1525 469 ...
## $ Threonine : num 184.9 198.3 110 376.1 64.1 ...
## $ Trigonelline : num 943.9 208.5 192.5 992.3 86.5 ...
## $ Trimethylamine.N.oxide : num 2122 639 1153 1451 172 ...
## $ Tryptophan : num 259.8 83.1 82.3 235.1 103.5 ...
## $ Tyrosine : num 290 167.3 60.3 323.8 142.6 ...
## $ Uracil : num 111 47 31.5 30.6 44.3 ...
## $ Valine : num 86.5 110 59.1 102.5 160.8 ...
## $ Xylose : num 72.2 192.5 2164.6 125.2 186.8 ...
## $ cis.Aconitate : num 237 334 330 1863 101 ...
## $ myo.Inositol : num 135.6 376.1 86.5 247.2 750 ...
## $ trans.Aconitate : num 51.9 217 58.6 75.9 98.5 ...
## $ pi.Methylhistidine : num 157.6 308 145.5 249.6 84.8 ...
## $ tau.Methylhistidine : num 160.8 130.3 83.9 254.7 79.8 ...
```

El siguiente paso fue estructurar los datos para introducirlos en nuestro contenedor. Se crea la matriz de conteo con todos los datos metabólicos, el objeto colData que contendrá la información de las muestras. Además de el metadata con la información del experimento. Para ello es necesario transponer los datos de los metabolitos, puesto que el contenedor SummarizedExperiment tiene una disposición tal que las columnas corresponden a las muestras y las filas a las medidas (en nuestro caso metabolitos). Generamos finalmente el objeto SummarizedExperiment con toda la información del experimento.

```
counts <- as.matrix(human_cachexia[, -1])
counts <- t(counts)
colData <- DataFrame(Muscle.loss = human_cachexia$Muscle.loss)

metadata <- list(
  general_information = "Successfully passed sanity check!",
  samples = "Samples are not paired.",
  groups = "2 groups were detected in samples.",
  data_values = "All data values are numeric.",
  missing_values = "A total of 0 (0%) missing values were detected."
)

se <- SummarizedExperiment(assays=list(counts=counts), colData=colData,
  metadata = metadata)
se
```

```
## class: SummarizedExperiment
## dim: 63 77
## metadata(5): general_information samples groups data_values
## missing_values
## assays(1): counts
## rownames(63): X1.6.Anhydro.beta.D.glucose X1.Methylnicotinamide ...
## pi.Methylhistidine tau.Methylhistidine
## rowData names(0):
## colnames(77): PIF_178 PIF_087 ... NETL_003_V1 NETL_003_V2
## colData names(1): Muscle.loss
```

```
dimnames(se)
```

```
## [[1]]
```

```

## [1] "X1.6.Anhydro.beta.D.glucose" "X1.Methylnicotinamide"
## [3] "X2.Aminobutyrate"             "X2.Hydroxyisobutyrate"
## [5] "X2.Oxoglutarate"             "X3.Aminoisobutyrate"
## [7] "X3.Hydroxybutyrate"           "X3.Hydroxyisovalerate"
## [9] "X3.Indoxylsulfate"           "X4.Hydroxyphenylacetate"
## [11] "Acetate"                      "Acetone"
## [13] "Adipate"                      "Alanine"
## [15] "Asparagine"                   "Betaine"
## [17] "Carnitine"                    "Citrate"
## [19] "Creatine"                     "Creatinine"
## [21] "Dimethylamine"               "Ethanolamine"
## [23] "Formate"                      "Fucose"
## [25] "Fumarate"                     "Glucose"
## [27] "Glutamine"                   "Glycine"
## [29] "Glycolate"                   "Guanidoacetate"
## [31] "Hippurate"                   "Histidine"
## [33] "Hypoxanthine"                "Isoleucine"
## [35] "Lactate"                     "Leucine"
## [37] "Lysine"                      "Methylamine"
## [39] "Methylguanidine"             "N.N.Dimethylglycine"
## [41] "O.Acetylcarnitine"           "Pantothenate"
## [43] "Pyroglutamate"              "Pyruvate"
## [45] "Quinolate"                   "Serine"
## [47] "Succinate"                   "Sucrose"
## [49] "Tartrate"                    "Taurine"
## [51] "Threonine"                   "Trigonelline"
## [53] "Trimethylamine.N.oxide"      "Tryptophan"
## [55] "Tyrosine"                    "Uracil"
## [57] "Valine"                      "Xylose"
## [59] "cis.Aconitate"               "myo.Inositol"
## [61] "trans.Aconitate"             "pi.Methylhistidine"
## [63] "tau.Methylhistidine"
##
## [[2]]
## [1] "PIF_178"      "PIF_087"      "PIF_090"      "NETL_005_V1"  "PIF_115"
## [6] "PIF_110"      "NETL_019_V1"  "NETCR_014_V1"  "NETCR_014_V2" "PIF_154"
## [11] "NETL_022_V1"  "NETL_022_V2"  "NETL_008_V1"  "PIF_146"      "PIF_119"
## [16] "PIF_099"      "PIF_162"      "PIF_160"      "PIF_113"      "PIF_143"
## [21] "NETCR_007_V1" "NETCR_007_V2" "PIF_137"      "PIF_100"      "NETL_004_V1"
## [26] "PIF_094"      "PIF_132"      "PIF_163"      "NETCR_003_V1" "NETL_028_V1"
## [31] "NETL_028_V2"  "NETCR_013_V1" "NETL_020_V1"  "NETL_020_V2"  "PIF_192"
## [36] "NETCR_012_V1" "NETCR_012_V2" "PIF_089"      "NETCR_002_V1" "PIF_179"
## [41] "PIF_114"      "NETCR_006_V1" "PIF_141"      "NETCR_025_V1" "NETCR_025_V2"
## [46] "NETCR_016_V1" "PIF_116"      "PIF_191"      "PIF_164"      "NETL_013_V1"
## [51] "PIF_188"      "PIF_195"      "NETCR_015_V1" "PIF_102"      "NETL_010_V1"
## [56] "NETL_010_V2"  "NETL_001_V1"  "NETCR_015_V2" "NETCR_005_V1" "PIF_111"
## [61] "PIF_171"      "NETCR_008_V1" "NETCR_008_V2" "NETL_017_V1"  "NETL_017_V2"
## [66] "NETL_002_V1"  "NETL_002_V2"  "PIF_190"      "NETCR_009_V1" "NETCR_009_V2"
## [71] "NETL_007_V1"  "PIF_112"      "NETCR_019_V2" "NETL_012_V1"  "NETL_012_V2"
## [76] "NETL_003_V1"  "NETL_003_V2"

```

```
colData(se)
```

```

## DataFrame with 77 rows and 1 column
##           Muscle.loss

```

```
##                               <factor>
## PIF_178                      cachexic
## PIF_087                      cachexic
## PIF_090                      cachexic
## NETL_005_V1                  cachexic
## PIF_115                      cachexic
## ...                          ...
## NETCR_019_V2                 control
## NETL_012_V1                  control
## NETL_012_V2                  control
## NETL_003_V1                  control
## NETL_003_V2                  control
```

A partir de nuestro objeto `se` podemos obtener la matriz de conteos con las concentraciones de los metabolitos para cada muestra. Además podemos filtrar los datos de metabolitos por grupo de muestras así como ver las distribuciones de las medias entre los mismos y hacernos una idea si existen muchas diferencias en los niveles.

```
counts <- assay(se)

mean_control <- rowMeans(counts[,se$Muscle.loss=="control"])
mean_cachexia <- rowMeans(counts[,se$Muscle.loss=="cachexic"])

mean_comparison <- data.frame(
  Media_Control = mean_control,
  Media_Cachexia = mean_cachexia
)
mean_comparison
```

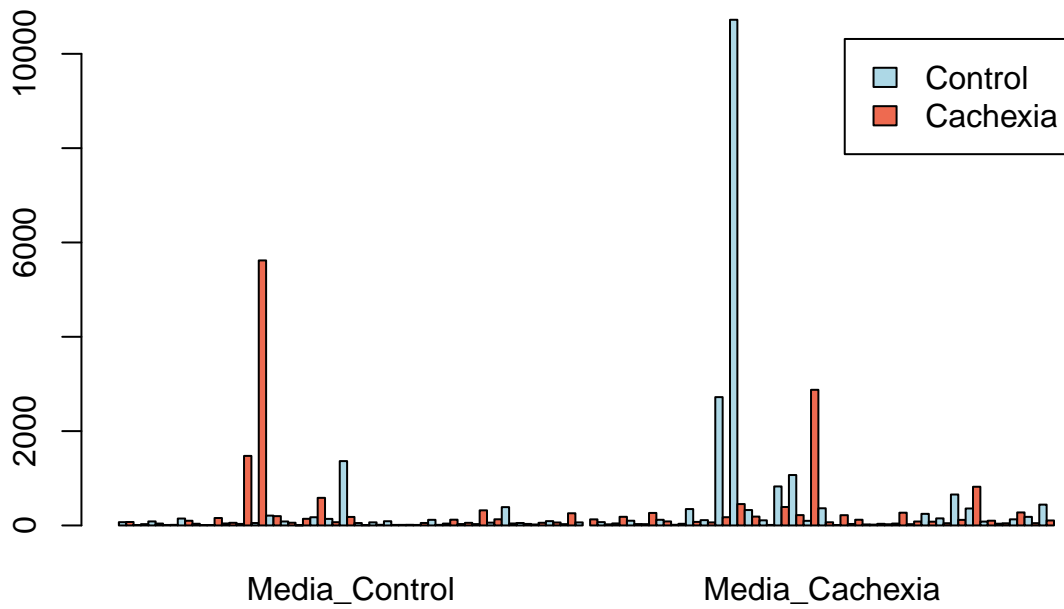
##	Media_Control	Media_Cachexia
## X1.6.Anhydro.beta.D.glucose	69.505333	128.688936
## X1.Methylnicotinamide	73.155000	70.564255
## X2.Aminobutyrate	9.528333	23.669149
## X2.Hydroxyisobutyrate	27.871000	43.237660
## X2.Oxoglutarate	85.517333	183.110426
## X3.Aminoisobutyrate	39.911000	100.274681
## X3.Hydroxybutyrate	9.898667	29.260638
## X3.Hydroxyisovalerate	12.312667	27.606383
## X3.Indoxylsulfate	146.376333	265.157660
## X4.Hydroxyphenylacetate	99.798667	119.822553
## Acetate	35.604667	85.632979
## Acetone	8.420000	13.346383
## Adipate	8.993333	34.817872
## Alanine	157.584000	347.591064
## Asparagine	41.749000	75.390851
## Betaine	55.970333	112.252979
## Carnitine	32.443667	64.622128
## Citrate	1474.718667	2720.852766
## Creatine	51.504333	174.913404
## Creatinine	5619.174667	10722.140213
## Dimethylamine	208.683333	453.580638
## Ethanolamine	197.125333	326.772128
## Formate	84.483333	187.564468
## Fucose	57.444667	108.599149
## Fumarate	4.552000	10.921915
## Glucose	140.958000	827.218936

## Glutamine	174.427333	391.410426
## Glycine	585.149333	1069.377872
## Glycolate	138.983667	219.269574
## Guanidoacetate	68.739667	97.624255
## Hippurate	1364.240333	2875.729574
## Histidine	180.472333	364.232340
## Hypoxanthine	51.714333	67.087021
## Isoleucine	7.218000	9.660851
## Lactate	65.748333	217.631915
## Leucine	13.556667	31.261702
## Lysine	89.229333	121.282340
## Methylamine	11.360000	21.216383
## Methylguanidine	12.128333	17.364681
## N.N.Dimethylglycine	13.596667	34.489787
## O.Acetylcarnitine	10.598000	25.564468
## Pantothenate	52.622667	39.944043
## Pyroglutamate	119.258000	270.292340
## Pyruvate	12.566333	26.865532
## Quinolate	39.324000	83.747234
## Serine	122.263000	245.829787
## Succinate	29.836000	79.628936
## Sucrose	55.579667	150.024468
## Tartrate	28.676000	47.234681
## Taurine	320.522333	655.720000
## Threonine	59.518667	118.233191
## Trigonelline	130.687000	359.637660
## Trimethylamine.N.oxide	388.669000	820.340638
## Tryptophan	41.833000	81.824043
## Tyrosine	52.014000	100.742340
## Uracil	32.493333	37.513617
## Valine	20.132667	45.582553
## Xylose	56.509333	129.289149
## cis.Aconitate	91.724000	276.025532
## myo.Inositol	62.641333	181.837660
## trans.Aconitate	27.809333	48.814043
## pi.Methylhistidine	258.640000	441.553191
## tau.Methylhistidine	64.650333	105.667660

```

barplot(as.matrix(mean_comparison),
        beside = TRUE,
        col = c("lightblue", "coral2"),
        legend = c("Control", "Cachexia"))

```



Exploración de datos

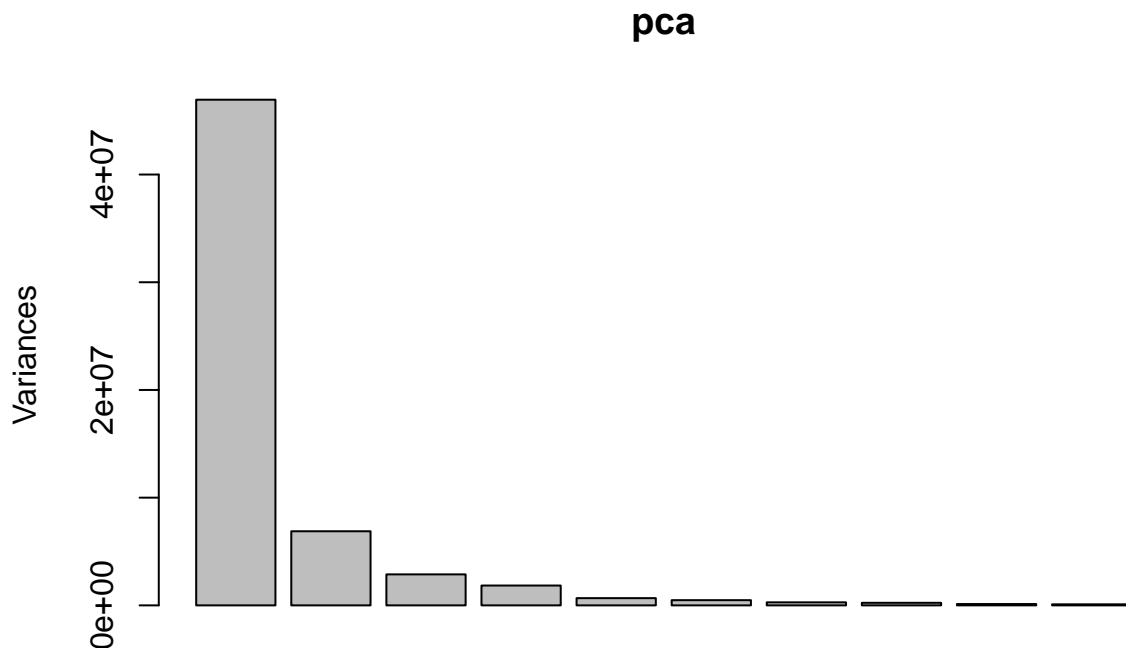
Posteriormente, se llevó a cabo un análisis de componentes principales (PCA) para observar la variabilidad de los metabolitos entre los grupos. Para ello usamos `prcomp()` explotando la primera columna que contiene la información de pertenencia a un grupo (ya sea controles o cachexia).

```
pca <- prcomp(human_cachexia[, -1])
sum_pca <- summary(pca)
sum_pca
```

```
## Importance of components:
##              PC1      PC2      PC3      PC4      PC5
## Standard deviation 6851.7365 2622.8132 1.696e+03 1.357e+03 819.83955
## Proportion of Variance 0.7717 0.1131 4.728e-02 3.027e-02 0.01105
## Cumulative Proportion 0.7717 0.8848 9.321e-01 9.623e-01 0.97339
##              PC6      PC7      PC8      PC9      PC10
## Standard deviation 695.64791 530.64795 487.0820 351.33316 309.39798
## Proportion of Variance 0.00795 0.00463 0.0039 0.00203 0.00157
## Cumulative Proportion 0.98134 0.98597 0.9899 0.99190 0.99347
##              PC11     PC12     PC13     PC14     PC15
## Standard deviation 247.94190 226.25912 213.79022 192.05209 162.75532
## Proportion of Variance 0.00101 0.00084 0.00075 0.00061 0.00044
## Cumulative Proportion 0.99448 0.99533 0.99608 0.99668 0.99712
##              PC16     PC17     PC18     PC19     PC20
## Standard deviation 155.7620 140.93059 138.48473 127.11181 120.82023
## Proportion of Variance 0.0004 0.00033 0.00032 0.00027 0.00024
## Cumulative Proportion 0.9975 0.99784 0.99816 0.99842 0.99866
##              PC21     PC22     PC23     PC24     PC25     PC26
## Standard deviation 119.07949 103.65002 99.67204 87.69537 81.35187 69.02493
## Proportion of Variance 0.00023 0.00018 0.00016 0.00013 0.00011 0.00008
## Cumulative Proportion 0.99890 0.99907 0.99924 0.99936 0.99947 0.99955
##              PC27     PC28     PC29     PC30     PC31     PC32
## Standard deviation 67.48679 63.70125 57.52296 50.81631 46.91029 46.28588
## Proportion of Variance 0.00007 0.00007 0.00005 0.00004 0.00004 0.00004
## Cumulative Proportion 0.99963 0.99969 0.99975 0.99979 0.99983 0.99986
```

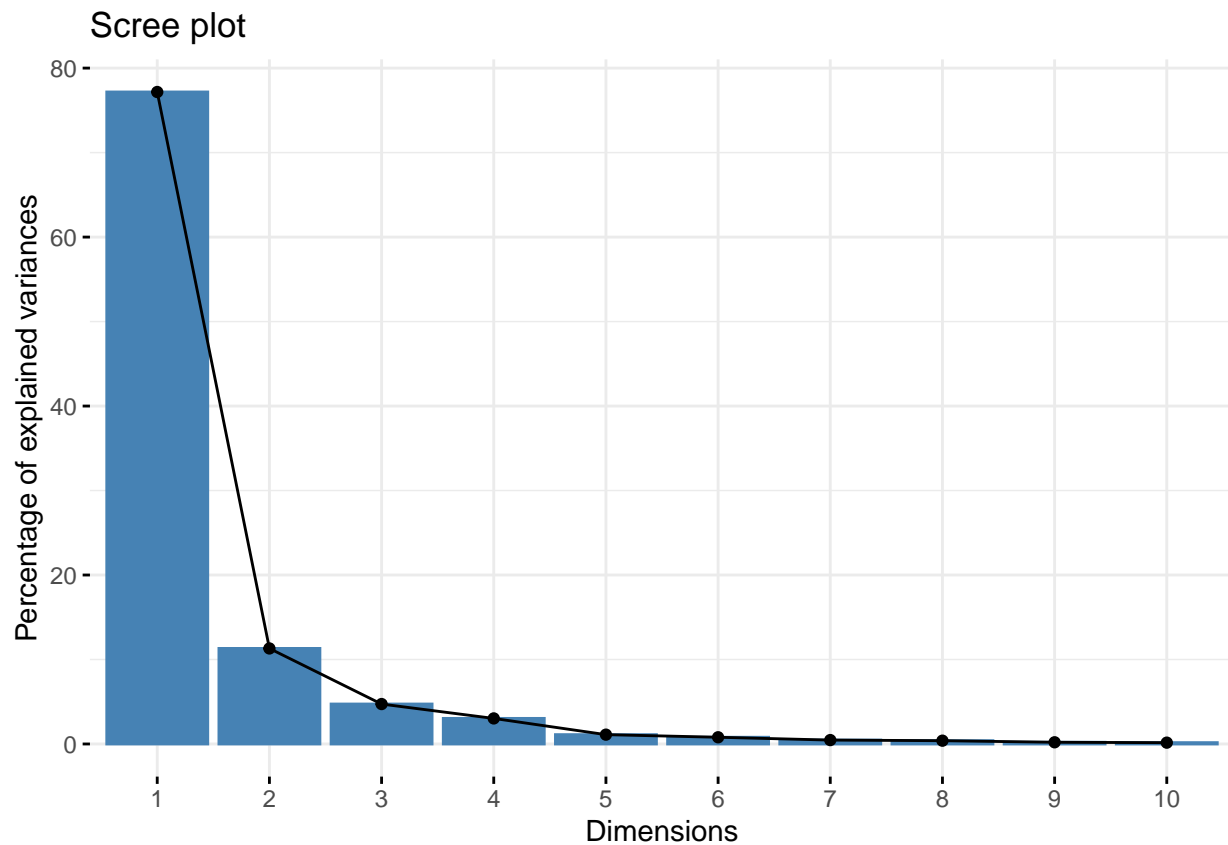
```
##          PC33      PC34      PC35      PC36      PC37      PC38
## Standard deviation 40.85547 36.52620 31.29715 28.31245 26.01032 22.69139
## Proportion of Variance 0.00003 0.00002 0.00002 0.00001 0.00001 0.00001
## Cumulative Proportion 0.99989 0.99991 0.99993 0.99994 0.99995 0.99996
##          PC39      PC40      PC41      PC42      PC43      PC44      PC45      PC46
## Standard deviation 21.72509 20.18788 17.62542 15.78 15.54 13.6 11.58 10.52
## Proportion of Variance 0.00001 0.00001 0.00001 0.00 0.00 0.0 0.00 0.00
## Cumulative Proportion 0.99997 0.99997 0.99998 1.00 1.00 1.0 1.00 1.00
##          PC47      PC48      PC49      PC50      PC51      PC52      PC53      PC54      PC55
## Standard deviation 9.081 8.383 7.52 6.72 6.077 4.932 4.047 3.618 3.272
## Proportion of Variance 0.000 0.000 0.00 0.00 0.000 0.000 0.000 0.000 0.000
## Cumulative Proportion 1.000 1.000 1.00 1.00 1.000 1.000 1.000 1.000 1.000
##          PC56      PC57      PC58      PC59      PC60      PC61      PC62      PC63
## Standard deviation 3.111 2.921 2.563 2.194 1.676 1.419 0.8066 0.3846
## Proportion of Variance 0.000 0.000 0.000 0.000 0.000 0.000 0.0000 0.0000
## Cumulative Proportion 1.000 1.000 1.000 1.000 1.000 1.000 1.0000 1.0000
```

```
plot(pca)
```



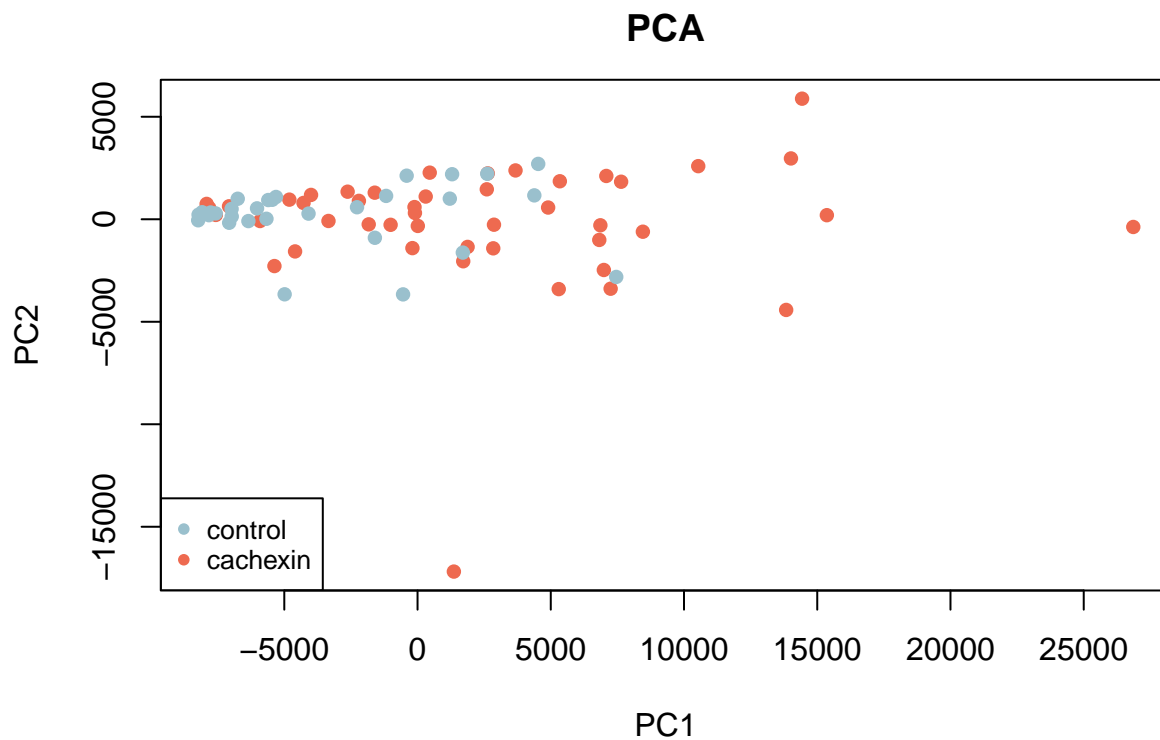
El resumen del PCA indica que los dos primeros componentes nos permiten explicar una proporción elevada de la varianza total por lo que pueden usarse para la diferenciación entre grupos. De hecho se visualiza la propoción de varianza explicada con `fvis_eig()` lo que nos permitió determinar que los cuatro primeros componentes explican casi la totalidad de la variabilidad.

```
fviz_eig(pca)
```

Posteriormente, se genera el gráfico para mostrar los resultados del PCA en las dos componentes principales coloreando cada grupo de un color:

```
plot(pca$x[,1:2], col=ifelse(human_cachexia$Muscle.loss == "control", "lightblue3", "coral2"),
     pch =16,
     xlab="PC1", ylab="PC2")
legend("bottomleft", pch=16, col=c("lightblue3", "coral2"),
      cex=0.8, legend = c("control", "cachexin"))
title(main="PCA", line=1)
```

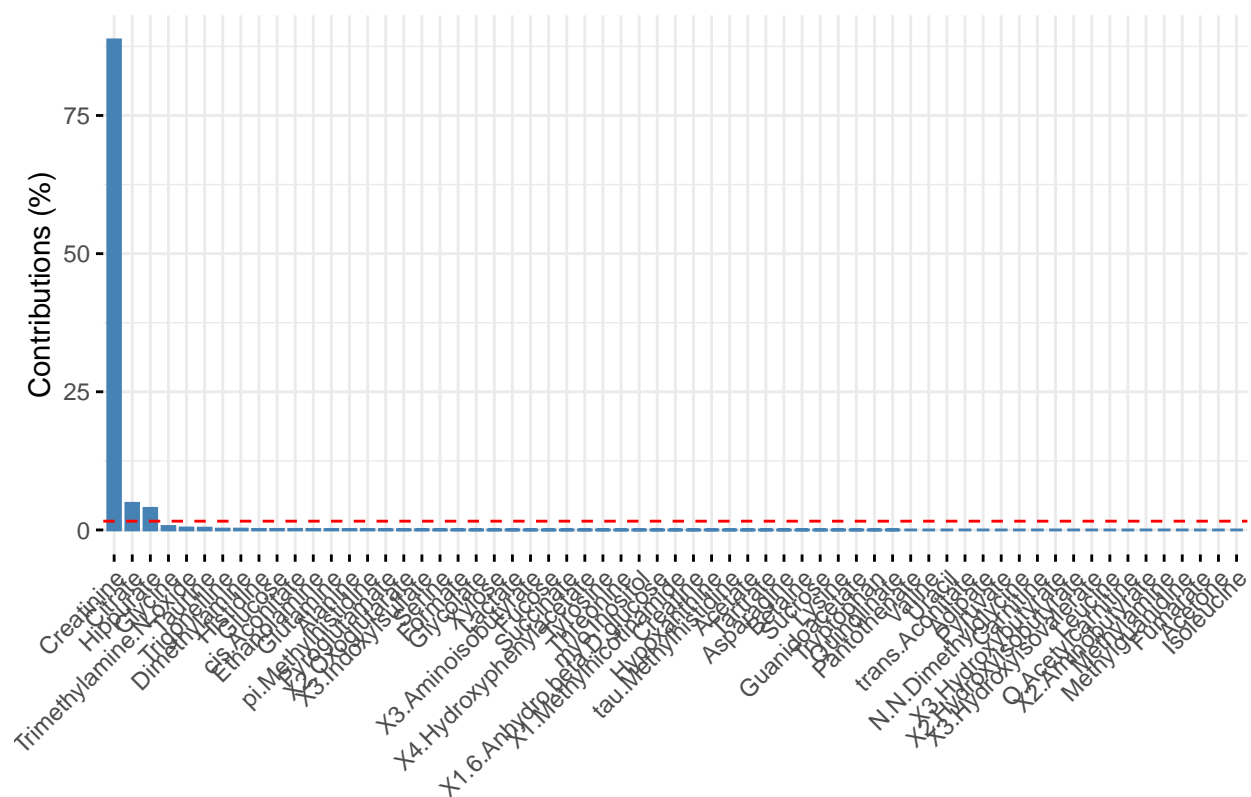


No observamos un patrón claro que separe nuestros dos grupos en poblaciones diferenciales.

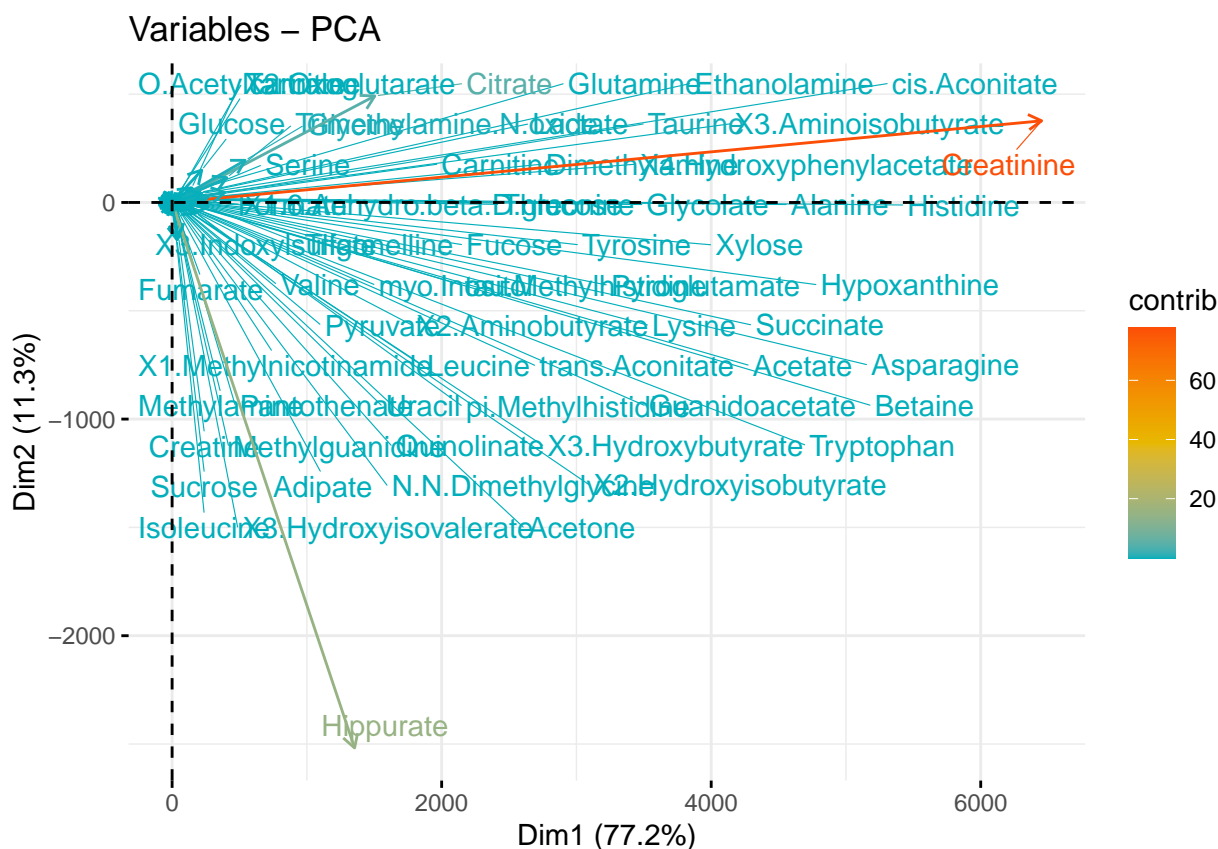
Para profundizar más en nuestro análisis y visto que en la gráfica que se generó con las medias entre grupos, parece existir variaciones en algunos metabolitos concretos evaluamos la contribución de las variables a los componentes principales:

```
fviz_contrib(pca, choice = "var")
```

Contribution of variables to Dim-1



```
fviz_pca_var(pca,
             col.var = "contrib",
             gradient.cols=c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE)
```



Tras este análisis observamos que creatina, citrato e hipopurato tienen una influencia mayor que el resto de metabolitos a la hora de determinar la distribución del PCA.

Discusión y conclusiones

Este análisis no ha permitido obtener una conclusión definitiva sobre los metabolitos específicos asociados a la pérdida de masa muscular. Sin embargo, se identificaron tres metabolitos con una influencia significativa en la distribución de los componentes principales, lo que sugiere que podrían ser biomarcadores prometedores para esta condición.

Para validar estos hallazgos, es fundamental realizar una serie de verificaciones iniciales. Primero, se debe revisar exhaustivamente los datos en busca de valores atípicos, ya que su presencia podría afectar la separación entre componentes y distorsionar los resultados del análisis basado en medias. Si los valores atípicos son descartados y las diferencias entre los niveles de estos metabolitos persisten, se recomienda un análisis más detallado enfocado en los componentes principales usando un conjunto reducido de metabolitos. Se podrían excluir aquellos metabolitos con poca o nula contribución en el análisis previo, y examinando si existe una correlación significativa entre los niveles de los metabolitos restantes y la presencia de debilidad muscular.

Repositorio de Github

<https://github.com/nugmi/Garcia-MartinezIllescas-Nuria-PEC1>